1. **Introduction**

This paper aims to address the technical approaches the team has used to develop the application, solutions to project management, and recommendations the team has for further development. The tools and technical approaches section will highlight the hardware and software used alongside development, and the project management section will detail how the team has encountered and learned from project management issues. The advice and recommendations section will contain personal recommendations from the team on what could be done differently for the next production iteration and advice for new teams that may work on this project.

The Primary tasks of this project are to deliver a battlefield experience in VR with a large focus on AI units. These AI units can be grounded or aerial, and will belong to one of three factions: Hostile, Allied, Neutral. The user will take the role of a commander and be able to issue orders to allied units to better help their team strategy. Hostile AI units will be given orders based on the data that can be gathered from the environment and their vision.

1. **Tools and Technical Approaches (about one paragraph per tool or technical approach)**

For this project, we utilized the virtual reality headset from Facebook called the Oculus Quest 2. This virtual reality headset allows for the running of the software on the headset itself independent of a computer in order to give it more mobility and less cumbersome as it would not have reh required cords. The unit itself uses a form of Android operating system in order to run the software on its device locally. This device, out of most virtual reality headsets is one of the more affordable but it is also somewhat restrictive in how it allows people to develop for it. One must first create a Facebook account and then a developer account with Oculus in order to setup the headset to allow for the usage of third party software.

For our editor software, used to develop everything from the game AI and the user interface we used Unity. A free software which allows for the creation of games in both 3D and 2D as well as various plugins which allow for the development of virtual reality ready games too. This software has a wide range of assets readily available for the user along with a more user-friendly development environment allowing for ease of use. We chose to use this software as it was at the request of our project partner.

In order to have our project be virtual reality capable, we at first utilized the Unity plugin developed by Oculus that allows for easy integration within a project. This product allows for a streamlined process in order to incorporate virtual reality controls and vision within a Unity project. We decided to make the move over to Steam VR as it is a more widely used plugin and it would allow for the usage of our software on more systems than those just running a Oculus Quest. This plugin is very fleshed out and well maintained with a active community making it very easy to make the transfer over.

1. **Project Management**

Our project was managed believing that there are three different areas of the project that all required a near-equal amount of work. The three areas are: Artificial Intelligence, Game Logic and Scene, and VR Visualization. The AI area would assume any tasks that related to the decision making of the non-player characters in the game. The VR Visualization assumed any tasks that were about displaying items in VR or handling input for VR. Then the third area, game logic and scene, was nearly a catch-all for any tasks that had to do with the operations of the game and the presentation of the scene. We didn’t officially choose who worked in what areas, but as we worked we slowly gravitated towards one person per area.

This initial division likely caused some slight issues with how we managed the project. One example is how we are still currently lacking the functionality required of a full virtual reality experience. While some of this is the fault of how we only have one headset, if we were to start over the project knowing what we know now, I would strongly lobby for more than one headset so that there may be some leakage in responsibilities when covering tasks related to virtual reality. Another issue that has arisen is how each individual person doesn’t have a huge amount of understanding of the other areas of the project because they have been only developing in one area.

The largest issue we ran into with our project management was specifically with our communication and interactions with our project partner, Raffael De Amacis. We struggled with ascertaining the exact requirements from our project partner and because of this we spent some time idling. This idling was because we weren’t entirely sure on what the project partner wanted from the project. While we would ask our project partner, and the questions were ignored, we should have just decided for ourselves so that the project could continue to be worked on. Towards the middle of winter term this was in fact what we started to do and with the new interactions with our secondary project partner we started to make much more serious progress on our project.

1. **Advice and Recommendations (about one paragraph per recommendation)**

Would recommend not using the Oculus VR plugin, and instead use a consistently maintained and current framework. Current roadblocks have led the team to migrate to the SteamVR plugin for development. SteamVR has documentation for both itself and its use with Unity. This will help expedite the process of setting up a VR set and developing inside Unity. The SteamVR plugin itself is more rigorously maintained and documented than the Oculus VR plugin, so there should be no issues with outdated dependencies and it should be easier to use.

A broad scope is the killer of projects. It is recommended to really focus the project to its core and to focus development on that. Incremental development tends to lend itself better to larger projects, so setting up the core of the application is important. By getting the core of the application finished, incremental updates can easily be tested and developed inside the core application.

It has been beneficial for the team to regularly meet and discuss ongoing development issues. These meetings give every member an opportunity to discuss current development issues, and discuss any overarching issues that may arise from a broader feature implementation. The meetings also give the team a chance to cement the direction of the project while preventing individuals from straying away from the project’s vision.